IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Validity	2. Conformance		3. Availability	4. Refreshment

1. PERCEIVED VALIDITY

The first criteria against which the reviewer weighs the effectiveness of the system is the users' perceived validity. Although completely subjective, it is the most important determinant of the system's effectiveness, because if the system users do not perceive the system as useful, accurate, and timely, the system will not be used and will degrade. To assess perceived validity, the reviewer does not rely on his/her own observations, but rather on the perceptions of both management and system users.

(1) Does the system provide the information/services expected by the user? (2) Does the user perceive the information provided by the system as accurate? (3) Does the user perceive the information provided by the system as complete? (4) Does the user perceive the information provided by the system as current? (5) Does the user perceive the information provided by the system as timely? (6) Does the user perceive the information provided by the system as adequate? (7) Does the user perceive the information provided by the system as consistent? (8) Does the user perceive the information provided by the system as concise? (9) Does the user perceive the information provided by the system as reliable? (10) Does the user perceive the information provided by the system as easy to use? (11) Does the user perceive the information provided by the system as important to performing his/her job?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Validity	2. Conformance		3. Availability	4. Refreshment

1. PERCEIVED VALIDITY

(12) Would a critical function not be accomplished if the system were not available to the user? (13) Would a critical function be delayed if the system were not available to the user? (14) Is all of the system information used?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Validity	2. Conformance		3. Availability	4. Refreshment

2. REQUIREMENTS CONFORMANCE

The depth of this part of the review will depend both on the availability of system requirements documents produced during development and how well they have been maintained during the system's life. The objective of this part of the review is to evaluate whether the system is accomplishing its established goals and whether those goals continue to be valid. The reviewer should address the achievement of both the functional and non-functional requirements.

(1) Does the system continue to meet the goals and objectives which prompted development? Does management periodically assess system performance against goals and objectives? (2) Do formal system requirements documents exist, technical and non-technical? (3) Are the original system requirements still valid? (4) Are there any existing system needs which are unmet? (5) Has the system fulfilled the expected cost/benefits defined in the system requirements? Has a recent evaluation been made? (6) Do the requirements address non-functional system goals? (7) Does the system adequately meet non-functional goals? (8) Are there pending environmental changes, such as new requirements or equipment purchases, that will affect the system requirements?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Validity	2. Conformance		3. Availability	4. Refreshment

3. SYSTEM AVAILABILITY

System availability for on-line systems is defined as the proportion of time the system is available to the user during normal working hours. Alternatively, if the system is a batch system, availability is the on-time delivery of scheduled output reports. The reviewer should determine whether system availability is affecting the users' ability to perform their jobs. If so, the reviewer should determine if the lack of availability is attributable to the hardware or software system.

(1) Is system availability an issue affecting users and/or the system objectives? (2) Are there system, performance, or maintenance logs available to evaluate availability or downtime? (3) What is the approximate percentage rate of system availability based on system demand? (4) What is the proportion of system downtime attributed to the hardware system? (5) What is the proportion of system downtime attributed to the software system?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Environment	E. Efficiency
1. Validity	2. Conformance		3. Availability	4. Refreshment

4. DATA REFRESHMENT

The area of data refreshment addresses the issue of how current the data provided to users is. This is in contrast to the perceived validity section where users' perceptions of data currency were established. The reviewer should confirm that data available to the user is adequately current; and if not, determine if there are procedures which could improve the timeliness of the information. To do so, the reviewer should discuss the data refreshment cycle with system maintenance personnel.

(1) Is the data refreshment cycle consistent with the data volatility? (2) Is data currency affecting functional requirements? (3) Can data currency be improved? (4) Is there a benefit to raising the data refreshment cycle?	NOTES

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. User Profile	2. General Interface	3. Documentation	4. User Support	5. Privacy

1. USER PROFILE

The user interface should be a function of user characteristics. Therefore, to evaluate the adequacy of the user interface, the reviewer must first develop a profile of the individuals using the system. Data reflecting relevant user attributes, such as education and technical experience, should be noted for the profile. The measure of user interface quality will be based on the user characteristics identified here. Profiles should be developed for each category of user, such as data entry and management.

(1) Classify the type of system users (e.g., clerical, analyst, managers). (2) Classify the experience level of system users (e.g., entry level, experienced, very experienced). (3) Quantify the turnover rate of system users (e.g., 5 replacements for a staff of 25 in the last year implies a 20% turnover). (4) Are some or all of the system users read-only users? Are read-only users prevented from writing (updating) the data? (5) Are some or all of the system users read/write users? Do passwords or other measures limit access to the write (update) functions? (6) Are the system users interactive users?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. User Profile	2. General Interface	3. Documentation	4. User Support	5. Privacy

2. GENERAL INTERFACE

Information can be presented to the user in many forms. Most often data is presented on hard copy output or through video display terminals. In this section, the questions posed are related to any form of communication between the user and the system. The term interface is used to refer to system input/output regardless of format. The reviewer should rate the user interface based on the user profile developed in the previous section.

(1) Is the interface appropriate to the system functional requirements? (2) Does the interface suit the needs of the intended user, in terms of device, media, and design? (3) Is the interface easy to access and use? (4) Is the interface nomenclature in terms familiar to the user? (5) Is the information presented concisely? (6) Is the interface suitable to the education and experience of the user? (7) Is the interface consistent across subsystems? (8) Does the interface allow ad hoc query and reporting? (9) Is the interface command, menu, or icon driven? Is it appropriate? (10) Are the help and error message functions sufficient and appropriate to the type of interface	NOTES:

and level of user? (11) Has color been used to advantage?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Environment	E. Efficiency
1. User Profile	2. General Interface	3. Documentation	4. User Support	5. Privacy

3a. DOCUMENTATION - Inventory

Regardless of the complexity or size of the system, the user must have access to information describing system capabilities, functions, and operations. This information may be in one document, a series of documents, on-line, or a combination of the foregoing. The reviewer should survey the information available to the user regardless of format. Reviewers should note that depending on the system, a system administrator's document may be required.

(1) Is there a functional description document defining the purpose of the system and what the system can do? (2) Is there an introductory manual which, in simple terms, explains how to log on the system, navigate through its functions, and handle problems? (3) Is there a comprehensive reference manual which describes in detail all of the system facilities available to the user and how these facilities can be used? Does the manual include a comprehensive index? (4) Is there a system administrator's guide explaining how to react to unusual situations and how to carry out housekeeping tasks? (5) Is the list of error messages complete, easy-to-find, and easy-to-use? (6) Is on-line help available? Is it documented and kept up-to-date?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Environment	E. Efficiency
1. User Profile	2. General Interface	3. Documentation	4. User Support	5. Privacy

3b. DOCUMENTATION - Quality

The quality of user documentation strengthens the user's ability to operate the system. Well organized, well written, and well maintained documentation makes users aware of the system capabilities and provides procedures to interact with the system. Reviewers will most likely gain their first knowledge of the system by reading user documentation.

(1) Is the user documentation current (i.e., reflects the last system change)? (2) Is the user documentation logically organized? (3) Is there a detailed table of contents in each document? (4) Are documents indexed to facilitate rapid location of specific items? (5) Is the document notation defined or readily apparent? (6) Has a standard procedure been used to layout the user documentation? (7) Can changed document pages be easily integrated into the document (section and page numbering)? (8) Are changed user documentation pages identified and dated? (9) Are changed paragraphs indicated to the reader?	NOTES:

(10) Is the user documentation style and format consistent from document to document?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Environment	E. Efficiency
1. User Profile	2. General Interface	3. Documentation	4. User Support	5. Privacy

3b. DOCUMENTATION - Quality

(11) Is the user documentation in good physical condition? (12) Is the user documentation clearly written such that the intended user can understand and exercise the system?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Environment	E. Efficiency
1. User Profile	2. General Interface	3. Documentation	4. User Support	5. Privacy

3c. DOCUMENTATION - Accessibility

To maintain the system, the maintenance staff must have direct access to all the system documentation. The reviewer should establish what documentation is available to the user and how easily the user can acquire needed information.

(1) Is there a sufficient quantity of user documentation? (2) Is the user documentation maintained in an automated form? (3) Is the user documentation on-line? (4) Has the user documentation been broadly disseminated to those needing it? Are updates and revisions broadly disseminated? (5) Are distribution lists maintained for updates and changes? Are updates and changes disseminated? (6) Is there a centralized master file of user documentation? Are obsolete documents retained in system archives? (7) Has an organization been assigned specific responsibility for maintaining and disseminating	NOTES:

system documentation?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Environment	E. Efficiency
1. User Profile	2. General Interface	3. Documentation	4. User Support	5. Privacy

4a. USER SUPPORT - Training

Good training is one of the most important factors in the successful operation of an automated system. In addition to helping attain the objectives of the system, it increases user confidence and raises job satisfaction. The reviewer's objective should be to determine if there is formal user training; and, if so, is it sufficient and appropriate to meet the needs of the user community.

(1) Is there a formalized ongoing training program? Does the program include classroom instruction, computer assisted training, training manuals, and/or individual instruction? (2) Is the training sufficient to develop the user skills required? (3) Is the training appropriate for the skill level of the trainees — whether directed at executive, managerial, technical, administrative, or clerical levels? (4) Does the frequency of training match the organization's turnover rate? (5) Are training facilities adequate?	NOTES:

(6) Are training aids adequate? (7) Are students tested or evaluated by supervisors on the training received? (8) Are trainers evaluated by the students? (9) Is there feedback to management on the effectiveness of training?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Environment	E. Efficiency
1. User Profile	2. General Interface	3. Documentation	4. User Support	5. Privacy

4b. USER SUPPORT - Technical Assistance

The operation of any large computer system requires that users have access to technical assistance. This can be in the form of a hot-line, on-site personnel, a help desk, or direct contact with the system developers and/or maintainers. By interviewing both technical assistance personnel and user staff, the reviewer should note whether the level of user assistance is suited to user needs and if the assistance is provided sufficiently, promptly, and courteously.

(1) Does a technical assistance function exist? (2) How is technical assistance made available to users? Are help desks, hot lines, user support offices, or bulletin boards used? How effective are they? (3) If the system is distributed, is help equally available to all system users? (4) Is technical assistance available during all work shifts? (5) Is the level of technical assistance appropriate to the needs of the users?	NOTES:

(6) Are technical assistance personnel sufficiently knowledgeable and experienced? (7) Are technical assistance personnel responsive and prompt? (8) Are technical assistance personnel proactive? (9) Is technical assistance available for all system aspects — such as equipment, operating and applications software, databases, communications, and so forth? (10) Is a technical assistance log maintained?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Environment	E. Efficiency
1. User Profile	2. General Interface	3. Documentation	4. User Support	5. Privacy

4b. USER SUPPORT - Technical Assistance

(11) What do help desk, trouble, and/or technical assistance logs reveal about the effectiveness of the technical assistance and the system in question? (12) Is there feedback from technical assistance staff to system maintenance staff? (13) Is there feedback from technical assistance staff to system training staff? (14) What perceptions are held by users about the effectiveness of technical assistance? What perceptions are held by technical assistance personnel about users and the system? (15) Is technical assistance being provided by system users not formally assigned the responsibility? Should the responsibility be assigned? Is technical support affecting the individual's job performance? Does the situation indicate ineffective or inadequate system technical support and/or training?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Environment	E. Efficiency
1. User Profile	2. General Interface	3. Documentation	4. User Support	5. Privacy

5. PRIVACY

The intent of this part of the review is to evaluate the data privacy and confidentiality aspects of the information system. The reviewer should make an evaluation of the system regarding access to and protection of personal and sensitive system data. Any potential problem areas should be brought to the attention of management.

(1) Is access to the system limited to those with a "need to know"?	NOTES:
(2) Are system security passwords required?	
(3) Are application security passwords required?	
(4) Are varying levels of user access employed to further data security?	
(5) Is access to system data limited to the functions required of the user or user organization?	
(6) Are separation of duties and checks and balances employed to protect against invasion of	

privacy and fraud, waste, and abuse? (7) Do work stations automatically log off if not keyed in a specific interval of time? (8) Do audit trails identify users accessing sensitive information? (9) Are personnel with access to personal or sensitive information screened?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	**2. Documentation**	**3. Action Type**	**4. Code Quality**	**5. Survivability**

1a. SYSTEM ENVIRONMENT - Hardware

The system environment encompasses all the facilities available to the software engineer for the maintenance and enhancement of the information system. For this review, the environment will include both the system and operational environments. The first system environment area is the hardware platform required by the information system. In addition to classifying the hardware, the reviewer is to assess the sufficiency and the expected life of the hardware system supporting the information system.

(1) What is the information system's architecture? Is the system based on a mainframe, minicomputers, or microcomputers? Is an inventory available? (2) Is system equipment within one generation of the manufacturers' current offerings? If not, is equipment obsolescence affecting system performance and reliability? Are plans underway for upgrade, enhancement, or replacement? (3) Is the hardware system capacity (storage, speed, etc.) sufficient for anticipated growth of this information system? (4) What proportion of hardware system workload is dedicated to this information system?	NOTES:

(5) When is it anticipated the hardware system will be replaced? (6) Does the information system require telecommunications? (7) How is telecommunications support provided (e.g., SNA, Ethernet, Novell)? (8) Is the capacity of the telecommunications network sufficient for anticipated growth?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability
1a. SYSTEM ENVIRONMENT - Hardware				
(9) Is the communications system dedicated to this information system? (10) Are there plans to enhance or replace the communications system?			NOTES:	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

1b. SYSTEM ENVIRONMENT - Software

The system software in this review includes the operating system, applications system, and, if used, the database management system (DBMS). The reviewer evaluates both the software currency and vendor support. This can best be assessed by comparing the software versions used to support the information system and the software vendor's latest offering.

(1) Classify the operating system supporting the information system (e.g., UNIX, MVS, DOS). (2) What version of the operating system is in use? (3) Is the operating system software more than six months behind the latest release offered by the vendor? (4) Is an operating system software upgrade anticipated within the next year that will affect the information system? (5) Are software upgrades tested and evaluated prior to implementation? Are opportunities for	NOTES:

productivity improvements requiring system maintenance actions evaluated? (6) What application language drives the information system? What version of the application language is used in the information system? (7) Is there more then one high level language used in the information system? (8) Is the application language version more than six months behind the latest version of the application language? (9) Is any assembly language used in the execution of the information system?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

1b. SYSTEM ENVIRONMENT - Software

(10) If applicable, specify the DBMS used by the information system. (11) Is the DBMS relational or hierarchical? (12) What version of the DBMS is used by the information system? (13) Is the DBMS version more than six months behind the latest release of the DBMS? (14) Is software currency or capability affecting system performance or reliability? Are plans underway for upgrade, enhancement, or replacement?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	**2. Documentation**	**3. Action Type**	**4. Code Quality**	**5. Survivability**

1c. OPERATIONAL ENVIRONMENT - Personnel

Those responsible for the maintenance of the system are critical to the system's success. Thus, there must be a sustained number of qualified people to support the system throughout its life cycle. The reviewer should evaluate whether the system's staff is sufficient in terms of numbers and education and/or experience. Further, the reviewer should assess whether staff familiar with the information system's operations are and will continue to be available to maintain the system over its life.

(1) Is the information system centralized or distributed? Are systems support personnel centralized or distributed? (2) How many maintenance personnel are assigned to supporting this information system? (3) How many maintenance personnel are available to support this information system and have some familiarity with the system? (4) What is the typical length of time a support person has been assigned to this information system?	NOTES:

(5) What is the average service length of software maintenance personnel? Will upcoming retirements affect the support function? (6) Is a trainee program in place to encourage continuity of staff expertise over time? (7) What is the turnover history of the staff supporting this information system? (8) What is the average level of total software experience attained by the system support staff? Is technical expertise limited to the extent that system performance is affected? (9) Is software support a potential problem for this information system?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	**2. Documentation**	**3. Action Type**	**4. Code Quality**	**5. Survivability**

1d. OPERATIONAL ENVIRONMENT - Facilities

Given that an adequate number of support personnel are available, the reviewer should evaluate whether they are provided with appropriate tools and access to the tools and the system. The level of support facilities should be commensurate with the size, age, and complexity of the information system.

(1) Is the information system centralized or distributed? Are the system facilities centralized or distributed? (2) Is there a systems manual library readily available? (3) Is there a source program library readily available? (4) Is there a "mirror" system used for testing? (5) Is interactive testing available? (6) Can system maintenance be accomplished during normal shift hours?	NOTES:

(7) Are maintenance work stations readily available? (8) Are test data generators available? (9) Are file comparitors available? (10) Are there trace debugging packages available? (11) Is the debugging environment interactive?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

2a. DOCUMENTATION - Inventory

The effectiveness of system documentation is directly related to the maintainability of a system. However, no matter how well organized, well written, and well maintained documentation is, the resources are useless unless the user can quickly identify and locate the needed documentation. Therefore, a current inventory of documentation becomes essential.

(1) Is there an inventory of documentation which is complete and current (i.e., reflects the last system change)? (2) Does the inventory list documents and synopsize their contents? Does the inventory identify either how to obtain the documents or specify their location? (3) Is the inventory readily available to support staff and users? (4) Is the inventory's level of detail commensurate with the size and complexity of the system under review?	NOTES:

(5) Does the inventory include all system documentation? (See Exhibit II-4 on page II-10 for a list of systems engineering documents.)	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

2b. DOCUMENTATION - Quality

The quality of system documentation is directly related to the maintainability of a system. Well organized, well written, and well maintained documentation provides the maintainers with the ability to quickly identify, locate, and resolve system deficiencies. As the reviewer becomes more familiar with the system, the adequacy of documentation quality should become apparent.

(1) Is the documentation current (i.e., reflects the last system change)? Is it thorough? (2) Is the documentation organization consistent with the system structure? (3) Is there a detailed table of contents in each document? (4) Are documents well indexed, facilitating rapid location of specific items? (5) Is the document notation defined or readily apparent?	NOTES:

(6) Are the documents structured to lead the reader into more formal and detailed descriptions of the system? (7) Has a standard procedure been used to layout the documentation? (8) Is documentation updated with system changes? (9) Can changed document pages be easily integrated into the document (section and page numbering)? (10) Are changed documentation pages identified and dated? (11) Are changed paragraphs indicated to the reader?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	**2. Documentation**	**3. Action Type**	**4. Code Quality**	**5. Survivability**
2b. DOCUMENTATION - Quality				
(12) Is there a minimum of document cross-referencing used? (13) Is the documentation style and format consistent from document to document? (14) Is the documentation in good physical condition? (15) Is the documentation clearly written such that the average maintainer can understand and modify the system?			NOTES:	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

2c. DOCUMENTATION - Accessibility

To maintain the system, the maintenance staff must have direct access to all the system documentation. The reviewer should establish both what documentation is available to the maintainer and how easily the maintainer can acquire needed information.

(1) Is the system documentation readily available to the maintenance staff? (2) Are all the documents centrally located in the maintainers' work area? (3) Are duplicate copies of documentation available? (4) Are master copies (paper and electronic) separately filed and accurately maintained? (5) Is the documentation maintained in an automated form?	NOTES:

(6) Is the automated documentation maintained in a word processing format? (7) Is the automated documentation maintained in a text editing format? (8) Is computer access to the automated documentation in a language commonly used by the maintainers? (9) Are current source code listings routinely maintained?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

3a. MAINTENANCE ACTION - Control Process

There are three categories of software maintenance: perfective maintenance, adaptive maintenance, and corrective maintenance. However, before looking at the types of maintenance performed on the system, the reviewer should evaluate the maintenance action control procedures.

(1) Are there procedures in place for users, management, and maintainers to convey suspected errors or proposed improvements to the maintenance organization? (2) Are requests for changes formally submitted? (3) Are maintenance actions logged? (4) Are maintenance actions prioritized?	NOTES:

(5) Is there a central point for all maintenance action approvals? (6) Are proposed maintenance actions reviewed by management? (7) Are maintenance actions scheduled? (8) Are maintenance actions categorized as perfective, adaptive, or corrective? (9) What are the proportions of perfective, adaptive, and corrective maintenance actions? Are costs and benefits for each maintenance action measured? (10) Are maintenance actions subjected to any cost/benefit review? (11) Are the configuration managers consulted before program changes are implemented?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability
3a. MAINTENANCE ACTION - Control Process				
(12) Are users consulted before program changes are implemented? Is notification of change disseminated? Does the maintenance staff coordinate with the training and technical assistance staffs? (13) Is there a formal process mandated when performing maintenance actions? (14) Is the formal process observed when performing maintenance actions? (15) Are code structures and conventions maintained during maintenance actions?			NOTES:	

(16) Are system maintenance actions documented sufficiently? (17) Is maintenance given a priority over development? (18) Are "quick fix" or patch type maintenance actions avoided? (19) Are maintenance action testing procedures adequate? (20) Is there a "mirror" system available to test maintenance actions? (21) Is there a formal test plan to evaluate maintenance actions?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

3b. MAINTENANCE ACTION TYPE - Perfective

Perfective maintenance includes changes which improve the system without changing the system's functionality. Perfective changes can be initiated by the system users or the system maintainers. In most cases perfective maintenance is discretionary. It can include reformatting of output or re-coding to improve execution speed.

(1) How many perfective maintenance actions have been completed in the past six months? (2) How many staff hours have been applied to perfective maintenance actions? What are the estimated costs and benefits?	NOTES:

(3) How many perfective maintenance actions are backlogged? (4) What is the time lag between request and execution of perfective maintenance actions? (5) Are perfective changes the result of user requests? (6) Are perfective changes the result of system personnel requests?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

3c. MAINTENANCE ACTION TYPE - Adaptive

Adaptive maintenance is performed in response to changes in the system's environment. In reviewing the system's adaptive maintenance, anticipated changes to the environment, such as changing requirements or a new procurement of hardware, should be noted.

(1) How many adaptive maintenance actions have been completed in the past six months? (2) How many staff hours have been applied to adaptive maintenance actions? What are the estimated costs and benefits? (3) How many adaptive maintenance actions are backlogged?	NOTES:

(4) Are adaptive changes the result of changes in the data formats or file structures? (5) Are adaptive changes the result of changes in the hardware configuration? (6) Are adaptive changes the result of changes in the requirements? (7) Are adaptive changes the result of changes in the operating system software?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	**2. Documentation**	**3. Action Type**	**4. Code Quality**	**5. Survivability**

3d. MAINTENANCE ACTION TYPE - Corrective

Corrective maintenance is performed in response to system errors (e.g., design, logic or coding) which were not found during the system development phase or which result from prior faulty maintenance actions. These system errors are most critical because they can result in erroneous data. As the system ages, the trend for the number of corrective maintenance actions should be downward. An unusually high number of corrective maintenance actions may prompt the reviewer to recommend system restructuring.

(1) How many corrective maintenance actions have been completed in the past six months? (2) How many staff hours have been applied to corrective maintenance actions? What are the estimated costs and benefits?	NOTES:

(3) How many corrective maintenance actions are backlogged? (4) What is the time lag between identifying the need for corrective maintenance and beginning the corrective process? (5) Are corrective maintenance actions the result of design errors? (6) Are corrective maintenance actions the result of logic errors? (7) Are corrective maintenance actions the result of coding errors? (8) Is the trend for corrective maintenance actions downward? (9) Does the system require "cleaning up" maintenance? (10) Should the system be restructured?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

3e. MAINTENANCE ACTION - Recommendation to Restructure

If the maintainability review indicates the system is seriously flawed, the reviewer may recommend that parts or all of the system be redesigned and/or restructured. The reviewer should consider the following questions when performing the evaluation.

(1) Are there frequent system failures? (2) Is the system code over seven years old? (3) Is there an overly complex program structure and logic?	NOTES:

(4) Is the code written for outdated hardware? (5) Is the system running in emulation mode? (6) Are there very large modules or unit subroutines? (7) Does the system require excessive resources? (8) Are there hard-coded parameters that are subject to change? (9) Is there difficulty in keeping the system maintenance staff? (10) Is the documentation seriously lacking? (11) Are there missing or incomplete design specifications?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	**2. Documentation**	**3. Action Type**	**4. Code Quality**	**5. Survivability**

4a. CODE QUALITY - Internal Documentation

System maintainers spend equal amounts of time trying to understand program code and making code changes. The most reliable source of information about the program is its source code and comments. If a program is well commented, it is probably well constructed. Reviewers should rate the program based on how easily modules are understood by looking at the source code.

(1) Do program modules contain a header block of commentary needed to identify and understand the purpose/function of the module?	NOTES:

(2) Is the code liberally commented to define the purpose of code groupings? (3) Is the style and format of program commentary consistent in all program modules? (4) Are constants and variables defined in the program? (5) Are inputs and outputs clearly stated by parameter lists or comments? (6) Is a variable name dictionary available? (7) Is a variable cross-reference list available? (8) Are decision points and subsequent branching accurately described? (9) Are deviations from forward (downward) logical flow explained? (10) Are unusual termination conditions described?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

4b. CODE QUALITY - Code Structure

Modular construction is the hallmark of well designed and well coded computer programs. Good system architecture apportions code into small self-contained functionally unique components. Modular code is easy to maintain because it is easier to understand functions within the modules as well as interactions between modules. Reviewers should consider the program code structure with relation to its modularity of code functions.

(1) Is the code structured? (2) Is the program easy to modify (module cohesion)?	NOTES:

(3) Is the program easy to understand (module independence)? (4) In general, are program modules less than 50 code lines? (5) Are the modules and organization of the programs obvious? (6) Is execution control and information flow between modules readily apparent? (7) Are functions, subroutines, or macros used in place of repeated code sequences? (8) Is the programming style consistent within modules? (9) Is the programming style consistent among modules?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

4c. CODE QUALITY - Naming Conventions

The names of entities in a computer program should be closely related to, and when practical, identical to the real-world entities they represent. Program names should provide the reader semantic clues which help to understand what the code is intended to do. The program should contain no anonymous constants. Constants should be named and the program parameter driven. Reviewers should evaluate how understandable program functions are through naming conventions.

(1) Are names of objects in the program identical or closely related to the real world entities they represent?	NOTES:

(2) Are program labels explanatory? (3) Are the naming conventions consistent among program modules? (4) Is the program free of jargon and poorly defined terms and symbols? (5) Is the program free of synonymous operandi — using two operand names for the same variable? (6) Is the program free of ambiguous operandi — using one operand name for different purposes? (7) Are variables packed into data words?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

4d. CODE QUALITY - Programming

Programs based in software engineering techniques will use structured programming as a foundation. However, the larger the system, the more difficult it is to adhere to strict top down structures. Regardless of structure, reviewers should assess the system's language construct.

(1) Is the program written in a single high-order language? (2) Has the program been written in compliance with a standard?	NOTES:

(3) Are basic high order language features used and non-standard compiler extensions avoided? (4) Does the program provide for check-point restart capability? (5) Are procedures provided to recover from exception termination? (6) When fatal exceptions (errors) are encountered does the program "fail-soft" (i.e., is execution terminated in a controlled manner)? (7) Does the program display clear and useful error messages? (8) Does the code handle recoverable errors? (9) Does the program protect against potential undefined operations (e.g., division by zero)? (10) Does the program provide meaningful error diagnostics? (11) Are print switches included that can be turned on for debugging?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability
4d. CODE QUALITY - Programming				
(12) Are subscripts range tested before they are used? (13) Are index parameters range tested before they are used? (14) Are hard coded constants values (e.g., tax rates) avoided in the program? (15) Do numerical processes use sufficient precision? (16) Is program/module size constricted by hardware resources (e.g., disk space, core)? (17) Is all code reachable?			NOTES:	

(18) Are there transfers to all labeled statements? (19) Is there only one entry and exit point within a module? (20) Is the program free of any unfinished modules? (21) Is the program free of dummy subroutines? (22) Is there any embedded use of proprietary code compilers? If yes, will other compilers work? Describe. (23) Is there any embedded use of proprietary CASE tool language which requires the application / operation of that tool for the system to work?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	**2. Documentation**	**3. Action Type**	**4. Code Quality**	**5. Survivability**

4e. CODE QUALITY - Source Code Formatting

Typographic style or source code formatting affects a reader's code comprehension. White space and code groupings provide visual clues about the program structure. Whether automated or manual formatting was used, reviewers should determine if the visual structure of the code lends comprehension to program/module.

(1) Does the program make liberal use of blank lines, reserved word highlighting, and consistent paragraphing to enhance readability? (2) Have layout conventions been adhered to? (3) Are local declarations separated from procedures? (4) Is consistent indentation used to enhance readability?	NOTES:

(5) Are line indentations between 2 and 4 characters? (6) Are if, for, while, repeat, begin, and end statements indented properly?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	**2. Documentation**	**3. Action Type**	**4. Code Quality**	**5. Survivability**

5a. SURVIVABILITY - Backup Provisions

Without back-up there is no recovery from disaster. Nonetheless, the level of effort expended in the back-up function should be cost effective and commensurate with the risk and degree of harm that would be suffered through the loss of the system. The reviewer is to survey the back-up provisions and determine how quickly and easily the system could be brought back to an operational state if a disaster occurred.

(1) Have arrangements been made with vendors or other agencies for substitute hardware in the event of a facilities disaster? Are agreements written? (2) Are periodic reviews made of substitute sites to assure continued minimum system compatibility and capacity? (3) Is there a complete set of current application program software and documentation	NOTES:

maintained off-site? (4) Is data backup frequency adequate for the data refreshment cycle? (5) Is data backup performed on an established schedule? (6) Are at least three levels or generations of backup maintained? (7) Are <u>complete</u> sets of data backups maintained off-site? (8) Are off-site materials clearly identified and dated?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Environment	2. Documentation	3. Action Type	4. Code Quality	5. Survivability

<u>5b. SURVIVABILITY - Backup Documentation</u>

In a disaster recovery situation, a blueprint is needed to describe how the system is to be reconstructed. Application programs, data, and documents must be maintained and accessible at an off-site location. The reviewer should ascertain the availability and adequacy of recovery plans and procedures necessary in the event the system experiences some form of disaster.

(1) Is there a disaster recovery plan? (2) Is the disaster recovery plan maintained off-site? Do key personnel have copies? (3) Is the disaster recovery plan complete?	NOTES:

(4) Has any Federal agency reviewed the disaster recovery plan? If so, who, when, and what was the result? (5) Are logs maintained of all application programs, system software, and hardware? (6) Are lists of documents, books, manuals, and other references needed to operate the system maintained? Are copies of the critical documents maintained off-site? (7) Are lists maintained of vendors and other contract services required to operate the system? (8) Are systems personnel informed of and prepared for their responsibilities in recovery operations? (9) Has the disaster recovery plan been fully tested?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Configuration	**2. Telecommunications**	**3. CASE Tools**	**4. Data Base**	**5. Portability**

1a. CONFIGURATION - Management

Software systems, particularly large systems, have long lifetimes — averaging an estimated seven to eight years — and are normally subjected to many changes. Asserting and managing control over system change is critical to long-term system effectiveness. Reviewers should evaluate whether the configuration management process is appropriate for the system under review.

(1) Is there a formalized configuration management plan? (2) Is the configuration management plan adhered to? (3) Is the configuration management plan automated? (4) Are there automated procedures to track system changes?	NOTES:

(5) Does the current configuration status reflect precisely the current physical system? (6) Is there a procedure in place to rebuild the system from its components? (7) Is there a specific list of documents under configuration management control? (8) Is the configuration management plan auditable? (9) Has there been a configuration audit in the past six months? (10) Are remote site configuration controls adequate?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Configuration	**2. Telecommunications**	**3. CASE Tools**	**4. Data Base**	**5. Portability**

1b. CONFIGURATION - Identification

To control a system, its parts must be identifiable. Circumstances may require that several versions of the system be supported at the same time to accommodate different platforms at different sites. To control the configuration, each unique unit of the system, including documentation, must be identified. The review team should determine if the system's elements are identified at a sufficient level to control system changes and revisions.

(1) Is the configuration baseline defined? Are system baseline components identifiable? (2) Is each configuration item, component, element, and/or unit identified uniquely and dated for last update? (3) Are configuration entities defined in a hierarchical structure?	NOTES:

(4) Are there adequate configuration management procedures to identify and retrieve magnetic media? (5) Do related configuration management documents have related document names (i.e., a hierarchical naming convention)? (6) Are there adequate configuration management procedures to identify and retrieve documentation? (7) Is there a system version naming scheme in place? Have program versions been numbered logically? (8) Are system components reusable?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Configuration	**2. Telecommunications**	**3. CASE Tools**	**4. Data Base**	**5. Portability**

1c. CONFIGURATION - Change Control

Managing change processes ensures that modifications to the system are made in a controlled way, allowing the effects of the changes on the system to be predicted. In this section, the reviewers should evaluate if mechanisms are in place to manage system changes.

(1) Is there a Change Control Board? If so, what organizations are represented? (2) Are there specific steps required to request a configuration change? (3) Are requests for configuration changes formally submitted? (4) Has the authority to approve changes been specifically assigned?	NOTES:

(5) What are the review procedures used to evaluate a change request? (6) Are life cycle management procedures followed to implement changes? (7) Are maintenance actions scheduled? (8) Are configuration changes executed in modest steps? (9) Are system changes and documentation changes strictly tied together? (10) Has the documentation been altered to coincide with software changes? (11) How is the system tested after change implementation? (12) Is the configuration control database automated?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Configuration	**2. Telecommunications**	**3. CASE Tools**	**4. Data Base**	**5. Portability**
1c. CONFIGURATION - Change Control				
(13) Are CASE tools used for configuration management? (14) Is the level of configuration management appropriate to the system? (15) Is a database schema in place to record configuration information? (16) Are users and the training organization informed of changes in a timely manner?			NOTES:	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Configuration	2. Telecommunications	3. CASE Tools	4. Data Base	5. Portability

2. TELECOMMUNICATIONS

The purpose of this portion of the systems engineering review is to ascertain the telecommunications attributes of the system under review and to consider whether the State is planning a transition to an open systems environment. Reviewers should identify the physical communications characteristics, evaluate the sufficiency of the current system, and list potential access enhancements.

(1) What are the characteristics of the system's telecommunications resources and/or services? (2) Has the State considered or planned for a transition to an open systems environment?	NOTES:

(3) Does the system involve any wide area network (WAN) telecommunications? (4) Does the WAN system use a commercial carrier for communications? (5) Does the WAN system use the X.25 communications protocol? (6) Does the system use any proprietary WAN protocols? (7) Does the system involve any local area network (LAN) communications? (8) Does the system involve interconnected LANs? (9) Does the system use any gateways, bridges, or routers? (10) Does the system use an automated system manager?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Configuration	**2. Telecommunications**	**3. CASE Tools**	**4. Data Base**	**5. Portability**

3. CASE TOOL USAGE

Various types of CASE tools are available for system development and maintenance. Reviewers should identify any CASE tools used on the system and synopsize their functions. If CASE tools are not being used, the reviewer should ascertain why.

(1) Were CASE tools used in the analysis and design of the information system? (2) Were CASE tools used in the development of the information system?	NOTES:

(3) Were CASE tools used for testing and debugging the system? (4) Are CASE tools used for testing and debugging system modifications? (5) Are CASE tools used to maintain the system and user documentation? (6) Are CASE tools used to manage the system configuration?	

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Configuration	2. Telecommunications	3. CASE Tools	4. Data Base	5. Portability

4a. DATA BASE MANAGEMENT SYSTEM - No DBMS

The following group of questions should be used if the information system does not use a database management system. The questions suggest those attributes which are indicative of a potential DBMS solution. Positive answers to several of the questions may prompt the reviewers' recommendation for redesign or redevelopment, if it would be cost beneficial to do so.

(1) Is there data redundancy among system data files? (2) Have changes in system requirements necessitated file format restructuring? (3) Do multiple system users require overlapping data? (4) Does the system require different levels of data access authority? (5) Are there ad hoc requests for data not available from standard outputs? (6) Does the same data in different files require concurrence?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Configuration	2. Telecommunications	3. CASE Tools	4. Data Base	5. Portability

4b. DATA BASE MANAGEMENT SYSTEM - DBMS

This set of questions should be used if the system incorporates a DBMS. For every DBMS there should be an active Data Base Administrator (DBA) responsible

for the system's technical control. The reviewer should identify the DBA and evaluate the adequacy of the current data base system.	
(1) Identify the DBMS in use. (2) Is the DBMS version more than six months behind the vendor's latest release? (3) Is the database serving other systems? (4) Is the DBMS capacity sufficient for anticipated system growth? (5) Is system response time sufficient for current and anticipated demand? (6) Has a Data Base Administrator been assigned? What are the functions of the DBA? Are the functions regularly performed? (7) Is the DBMS serving users' needs satisfactorily? Are DBMS maintenance procedures periodically performed? (8) Does the DBMS of choice restrict the State's hardware and/or software options? If so, does the DBMS remain cost beneficial?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Configuration	2. Telecommunications	3. CASE Tools	4. Data Base	5. Portability

5. PORTABILITY

Measurement of system portability is dependent as much on the target system as it is on the source system. Although a thorough evaluation of system portability

cannot be ascertained without identifying the target system, reviewers can measure some aspects of portability independent of the destination system. This is particularly helpful if the reviewer is considering the potential of a system for transportability between States.	
(1) Is the hardware platform a current offering from a major manufacturer? (2) Is the application language a current ANSI standard? (3) Is the database system a current offering from a major manufacturer? (4) Are fourth generation languages avoided? (5) Does the system use modular construction? (6) Are non-standard language extensions avoided? (7) Are machine dependent parts of the system isolated?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Software	**2. Hardware**	**3. Inventory**	**4. Reporting**	**5. Technologies**
1. SOFTWARE EFFICIENCY				

Using system or code analysis to identify major resource consumers in a program is the best way to improve software system efficiency. Once the "hot spots" are identified, an informed decision can be made as to the costs or benefits of optimizing system performance. In the absence of any code optimizer, the reviewer can only postulate as to the software efficiency.

(1) Is software efficiency a system issue? (2) Has a code optimizer been used on the system? (3) Has the optimizer identified any areas where it is cost/beneficial to tune the code? (4) Is the code optimizer rerun periodically and/or after all maintenance actions? (5) Does the source code appear to be coded for efficiency?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Software	**2. Hardware**	**3. Inventory**	**4. Reporting**	**5. Technologies**

2. HARDWARE EFFICIENCY

The best way to evaluate a system's workload is to examine it. Audit packages provide a picture of the system's real workload without added cost and automatically collect and present the information. Additionally, this data can also provide the reviewer with a statistical count of the resources used by the system.

(1) Is hardware efficiency a system issue? (2) Does the system include an audit package to capture performance indicators and system diagnostics? (3) Does the computer center management monitor system performance data on a regular basis? (4) Is there periodic control system tuning? (5) Does the hardware system have capacity for all applications? (6) What proportion of system processing, storage, communications, and peripheral devices are used to support the information system being reviewed?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Software	2. Hardware	3. Inventory	4. Reporting	5. Technologies

3. HARDWARE INVENTORY

An inventory of the hardware required to support the system should be developed if one is not already available. The objective of this part of the review is to determine if enough hardware is available to support the mission. In addition, the reviewer should note if excess unused equipment is on hand.

(1) What processors, storage devices and capacities, communications facilities, and other peripheral hardware are associated with the system? (2) Is sufficient hardware available to accomplish the mission efficiently? (3) Has excess equipment been eliminated and made available for other uses?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency

1. Software	2. Hardware	3. Inventory	4. Reporting	5. Technologies

4. EFFICIENCY REPORTING

The objective of this section is to evaluate the reporting of operating system efficiency to management. The system and content of the reporting typifies the quality of operating system efficiency. The reviewer should survey operating system reports to evaluate system performance.

(1) Is hardware system performance data communicated to upper management? (2) Is the information reported on a regular basis? (3) Is the content of the report in terminology relevant to management perspectives? (4) Do the reports enhance the management process? (5) Do the reports objectively reflect performance? (6) Do the reports relate to goals and objectives, the budget process, and procurement planning? (7) Are actions taken in response to the reports?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency

1. Software	2. Hardware	3. Inventory	4. Reporting	5. Technologies

5a. TECHNOLOGIES - Open Systems Environment

A major data processing objective for government and industry is migration to an open systems environment (OSE). The reviewer should examine the means by which the State considers, plans, and makes decisions relating to and affecting system interoperability, portability, and migration to open systems.

(1) Does the State monitor developments in industry standards and evaluate their relevance to State systems, current and future? (2) Are decisions made in a manner which lessens the State's dependency on proprietary vendor solutions in favor of a reliance on industry standard protocols? (3) Does the State plan an evolution to an open systems environment? What processes and procedures support the goal? (4) Does the State adopt formal industry standards [such as ISO's Open Systems Interconnect (ISO) architecture] and/or de facto industry standards [such as Transmission Control Protocol / Internet Protocol (TCP/IP]? (5) Is the operating system UNIX or POSIX based? (6) Does the database management system use a Standard Query Language (SQL) interface? (7) Is the applications programming language C, COBOL, Fortran, Ada, or Pascal?	NOTES:

IV. SYSTEM ENGINEERING AND ARCHITECTURAL REVIEW

A. Effectiveness	B. User Interface	C. Maintainability	D. Attributes	E. Efficiency
1. Software	2. Hardware	3. Inventory	4. Reporting	5. Technologies

5b. TECHNOLOGIES - Software Reuse

The maturation of structured coding techniques has led the way for reuse of software from all phases of the life cycle. The reviewer should seek software for reuse and promote the development of software which will be reusable (transportable) regardless of its development stage.

(1) Does the information system contain any reused software? (2) Does the information system have any software for potential reuse (e.g., functions common to other system processes)? (3) Is the potentially reusable software easily understood? (4) Is the potentially reusable software well documented? (5) For systems in the development stage, has software reuse been thoroughly evaluated?	NOTES: